#15 days dsa Coding challenge

# day 10

#Leetcode 111

#Minimum depth of binary tree

Solution:

/\*\*

\* Definition for a binary tree node.

\* struct TreeNode {

\* int val;

\* TreeNode \*left;

\* TreeNode \*right;

\* TreeNode() : val(0), left(nullptr), right(nullptr) {}

\* TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}

\* TreeNode(int x, TreeNode \*left, TreeNode \*right) : val(x), left(left), right(right) {}

\* };

\*/

class Solution {

public:

int minDepth(TreeNode\* root) {

if (!root) return 0;

if (!root->left && !root->right) return 1;

if (!root->left) return 1 + minDepth(root->right);

if (!root->right) return 1 + minDepth(root->left);

return 1 + min(minDepth(root->left), minDepth(root->right));

}

};